



ElvisPOS



MEMPHIS
PLAY AS YOU LIKE

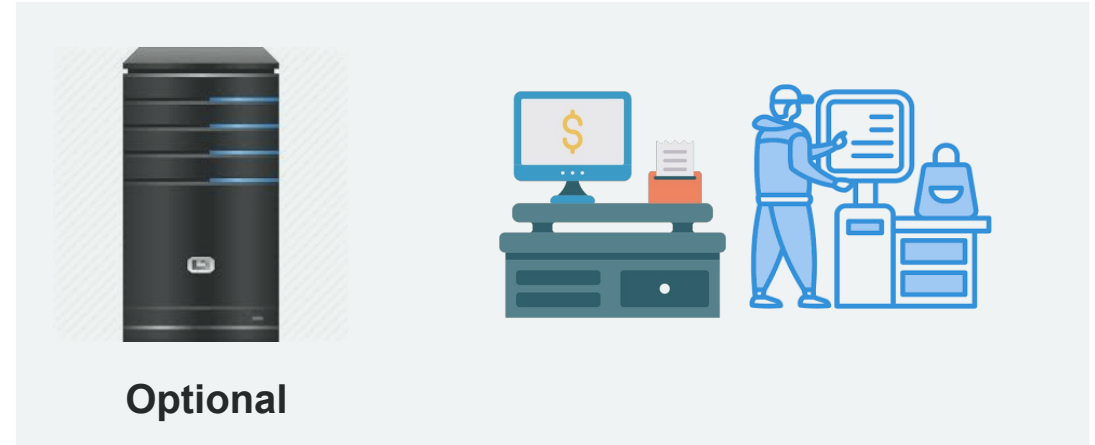
Architecture Overview

Agenda



- ✓ System Overview
- ✓ Architecture principles
- ✓ High level architecture
- ✓ Store architecture
- ✓ Central platform
- ✓ Data flow
- ✓ Integration architecture
- ✓ Security
- ✓ Scalability
- ✓ Deployment
- ✓ Software distribution
- ✓ Q&A

System Overview



Optional



MEMPHIS
PLAY AS YOU LIKE

- ✓ Organization Structure
- ✓ Parameter
- ✓ Software distribution
- ✓ Real time Transaction
- ✓ Web Reporting
- ✓ ERP - Back Office - BI Interface

- ✓ Sales process
- ✓ Store Operation
- ✓ Local parameters
- ✓ Local reporting

Architecture Principle



Built on **Open Source stack** based on **Robust components** and **Modern Programming Language**.

Deployed under **Docker** to guarantee scalability

Business Continuity its our compass on each technical decision:

- **Distributed DB** and **logic** at **POS** level (offline protection)
- **Sanity check** on data IN to guarantee data integrity
- Controlled **Software Distribution**

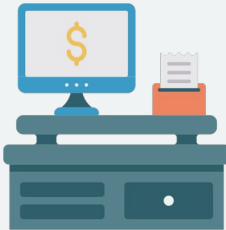
High Level Architecture

Common Stack

Device
Manager (JHIM)

Transaction
manager

Promo
Engine



Parameters

Software
distribution

Transaction
collection

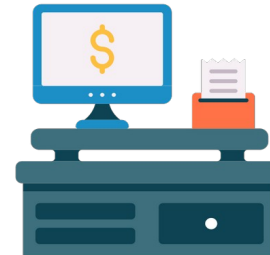
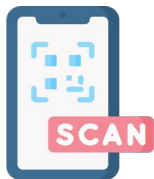
IN/OUT
Interface

Store Architecture

Data & Logic are distributed at POS, SCO and kiosk to guarantee **offline functionality**.

In-Store Server its optional and it can be used when:

- Back Office is not centralized or web
- POS network isolation its required
- Reporting or invoicing its required also when offline
- Cart calculation is required by third parties

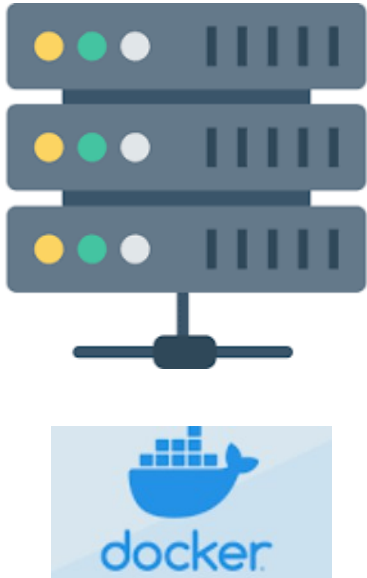


Back
Office



Central Platform

Head Quarter



It is the centralized console to defined organization structure, parameters, Stores and device configurations.

Data exchange with the store is done in real time.

It exposes Standard API and Message Queue to interact with ERP, Back Office, BI etc

Deployed under Docker for scalability and Software update

Within Memphis components bi-directional data synchronization its guarantee by **REBUS**

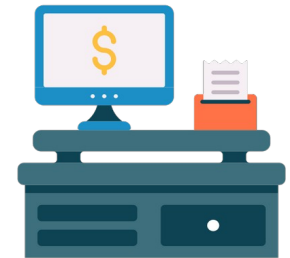
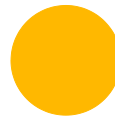
ReBus : Retail Bus



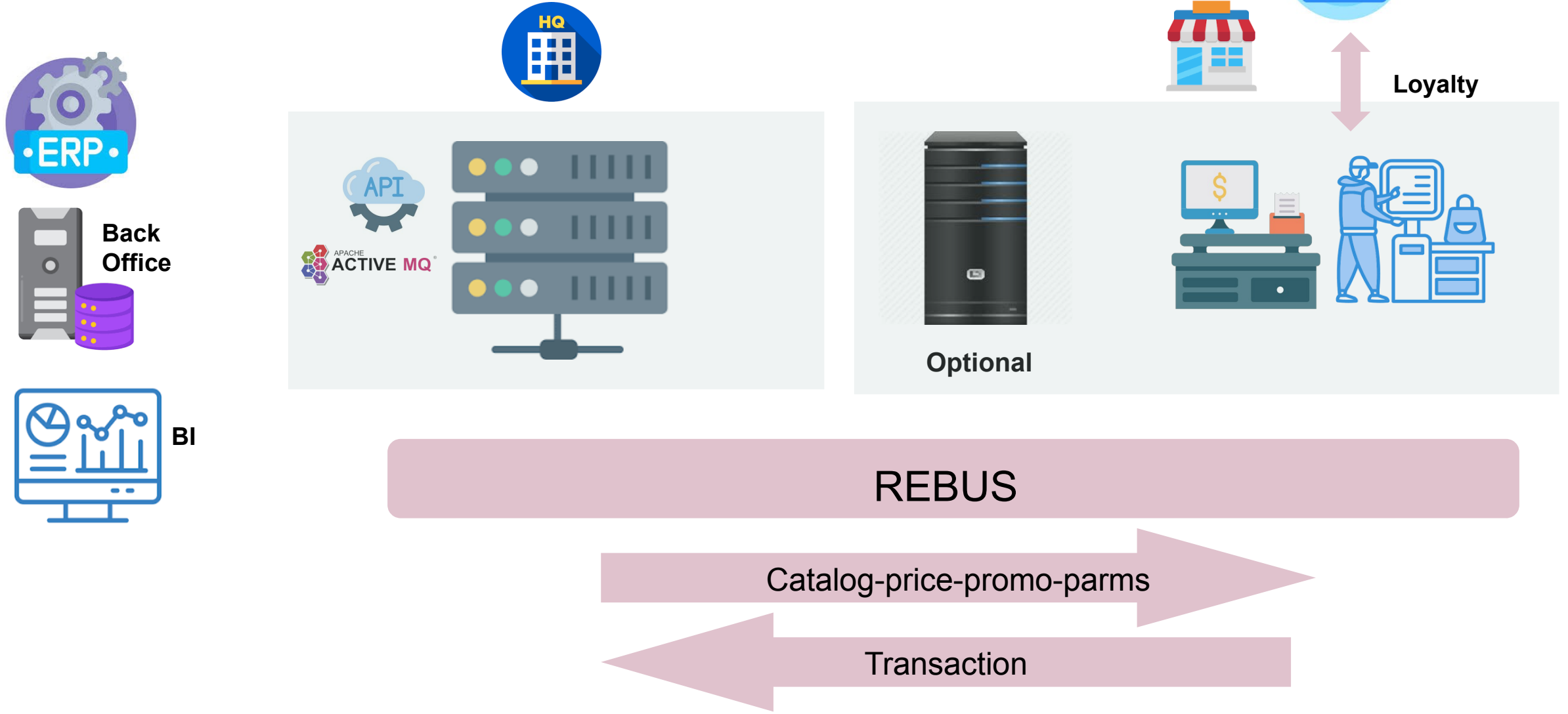
Developed in Java, it uses hybrid technology (*MQTT and Web Services*).

Each YELLOW Point communicates with its RED Point.

Sends transactions through queues (MQTT) and aligns the local database in "pull" mode using WS.



Data Flow

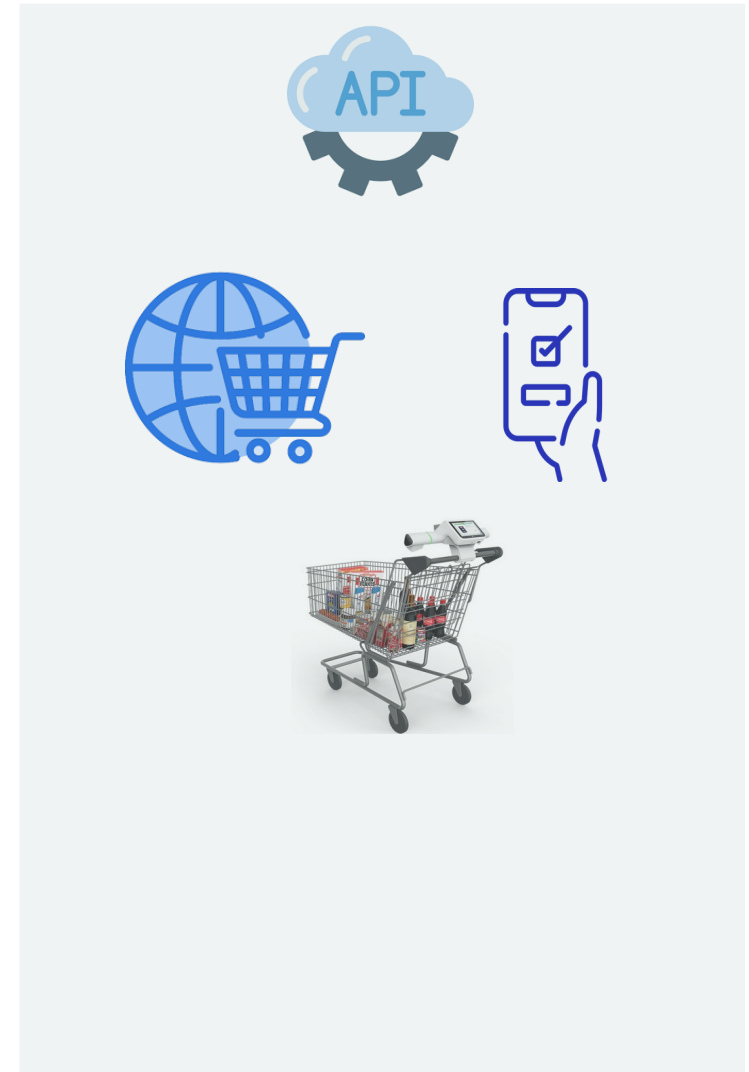
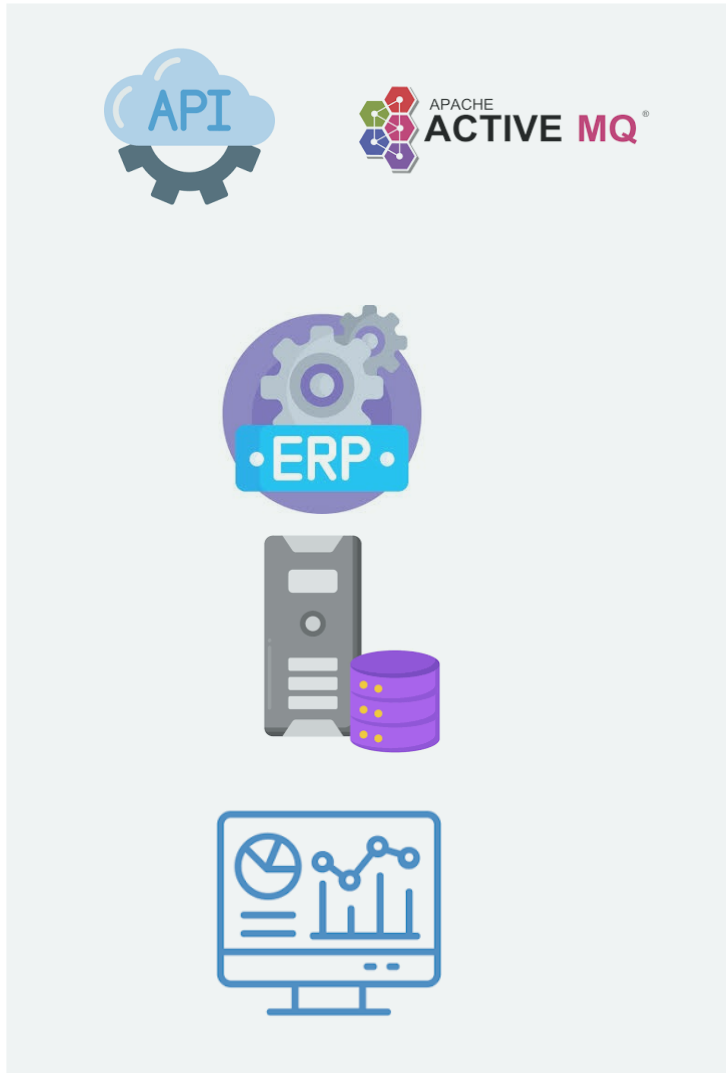


Integration Architecture

Retailer HQ Application

POS Peripherals

Third party application



Four API categories

Memphis export the following API categories:



- 1) FEED** the Memphis with data (Catalog, Prices, Promotion, groups...)
- 2) RESEND** via Message Queue a specific transaction period
- 3) CALCULATE** price and discounts of a cart (cart can be parked and resumed from any touch point)
- 4) PRICE INQUIRY** utilized by third parties Price Checker

Security



Normally Retail application are running in a protected network.

Following indication of a Third Party Audit we are working on the following aspects:

- Remove all the JAR files deprecated
- Update components to the latest available version
 - Artemis 2.37
 - commos-collections to 4.4+
 - upgrade Gson library to 2.8.9+
 - enforce SSL/TLS acceptors
 - update to jetty 12
 - update to Java 17

API access are provided via token authentication.

Artemis supports communication encryption that we currently are not using considering it over killing.

Scalability



As described, the POS has all the **data** and **logic** deployed locally.

A large number of POS connected to the same server might delay local price update or transaction collection but it will not impact sales performance.

In any case, the architecture allow Scalability strategy depending on the bottleneck:

processing speed (CPU or Memory) can be solved distributing Docker container on more servers

network traffic can be reduced installing Store servers on the bigger stores

Deployment

1

Configuration

- Company, logo, store
- Device type and peripherals
- Define Department, tender, VAT....



2

Store Server Setup (if any)

- install Debian ISO
- install ecli package
- run ecli setup with store # as parm



3

POS & SCO Setup

- Boot from Installation USB key
- Assign IP
- Enter server IP and Device ID

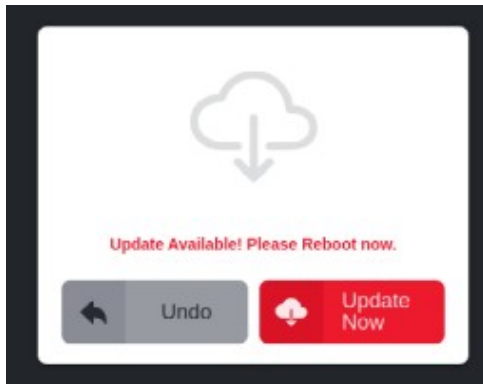


Software Distribution



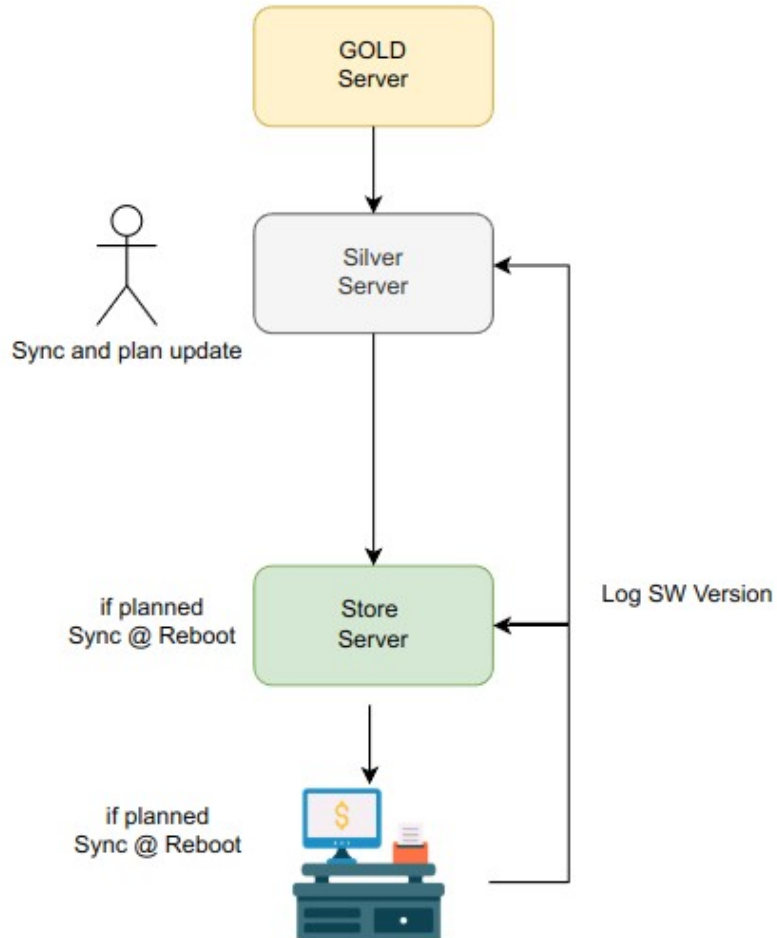
New versions are made available by Elvispos on the GOLD machine.

A dashboard allows to determine which nodes to update (test lab, pilot, and then rollout) while having complete visibility of the installed versions.



	Id	Server da Aggiornare	Aggiorna Casse Dispari	Aggiorna Casse Pari	Stato Negozio	Azienda	Luogo	Negozio	Totale Dispositivi	Disp. NON Allineato	Disp. da Aggiornare	Server Allineato	Disp. Sconosciuto
		<input type="checkbox"/> Seleziona Tutti	<input type="checkbox"/> Seleziona Tutti	<input type="checkbox"/> Seleziona Tutti	Q ↑	Q	Q	Q	Q ↑	Q ↑	Q ↑	Q ↑	Q ↑
1		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVE			ISO	1	1	0	No	0
2		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVE			ARDELLO	1	1	0	No	0
3		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVE			ANA	1	1	0	No	0
4		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVE			IA CLAUDIO	1	1	0	No	0
5		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVE			RS	2	2	0	No	1
6		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVE			UTI S.R.L.	2	2	0	No	0

Software distribution process



The SILVER server updates itself with the information published on the GOLD server using the manually executed ecli sync command.

The dashboard allows you to schedule the update of a store server and the master checkout.

The appropriate table indicates for which node the software update is available.

During the nightly reboot, the software maintenance procedure sequentially aligns:

- ✓ Datamodel
- ✓ Parameters
- ✓ Software

The node notifies you via the transaction log that the alignment has been completed.



Prepararsi sui temi fiscali

Spiegare come si lavora in team remoti

Integration:

- articoli e promo
- vendite